Imperceptible Data Transmission

U. Mourya Vardhan[1], A. Srinivasa Rao[2]
Department of Electronics and Communication[1, 2],
Nirma College of Engineering and Technology[1, 2],
Vijayawada, Andhra Pradesh, India[1, 2]

Abstract— **Steganographic is a art or method or technique of sending hidden data or secret messages over a public channel so that a third party cannot detect the presence of secret message. There are numerous proposed protocols to hidden data in channels containing pictures, video & audio and even typeset text. In this paper, secret communication through audio, i.e., embedding textual information in an audio file steganography. Through this technique the perceptual quality of the host audio files not degraded and not be detected.**

Keywords- Steganography; Human Auditory System (HAS); Hidden object; Hidden data; Stego-object; Embed, Extraction.

I. INTRODUCTION

Steganography is an art of sending hidden data or secret messages over a public channel so that a third party cannot detect the presence of the secret messages. The goal of steganography is different from classical encryption, which seeks to conceal the content of secret messages; steganography is about hiding the very existence of the secret messages. Modern steganography is generally understood to deal with electronic media rather than physical objects. There have been numerous proposals for protocols to hidden data in channels containing pictures [1-3] video [3, 4], audio [1, 3] and even typeset text [1, 3]. The size of the information is generally quite small compared to the size of the data in which it must be hidden (hidden text), electronic media is much easier to manipulate in order to hidden data and extract messages. The extraction itself can be automated. The main goal of this paper was to find a path to an audio file , a host media to hidden textual message without affecting the file structure and content of the audio file. No degradation in perceptual quality is the main objective of steganography.

II. ASSUMPTION AND SCOPE

Modern steganography based on embedding of secret data into electronic media like image [1, 2, 3], audio [1, 3], video [3, 4] and text [1, 3]. For example, to a computer, an image is an array of numbers that represent light intensities at various pixels. A common image size is 640 × 480 pixels and 256 colors (or 8 bits per pixel). Such an image could contain about 370 kilobits of data [5]. Digital images are typically stored in either 24 bit or 8 bit files. A 24 bit image provides the most space for hiding information. An image can be used as many times as possible by applying different types of filters or masking. A data – embedding technique into an audio file can be based on frequency masking [6], temporal masking [7], bit modification [8], LSB based method based on lifting wavelet transform [9] etc. while embedding text into an audio file least significant bit (LSB) not modification as it can create changes in the host audio file.

A steganography system, in general, is expected to meet three key requirements, namely, imperceptibility of embedding, accurate re-hiddendancy of embedded information, and large payload (payload is the bits that get delivered to the end user at the destination) [1]. A pure steganography framework technique be acts as a black box i.e., only between the sender and the receiver. The desired characteristics of an effective steganography are as follows:

Secrecy: No person cannot be able to extract hidden data from host file without the knowledge of secret key used in extraction process.

Imperceptibility*:* the medium after being embedded with the hidden data should be indiscernible from the original medium. One should not have the knowledge or suspicious about existence of the hidden data within the medium.

High capacity: The maximum length of the hidden message that can be embedded should be as long as possible. It depends on the size of the host audio file.

Resistance: The hidden data should be able to survive when the host medium has been manipulated, for example by some lossy compression scheme [12] and noise created by medium.

Accurate extraction: The extraction of the hidden data from the medium should be accurate and reliable.

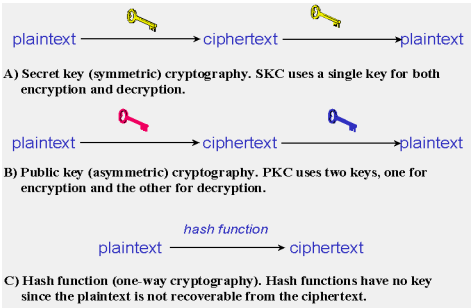

Figure 1. Cryptography

Basically, the purpose of steganography is to provide secret communication like cryptography. In cryptography, the system is broke down when the hacker can read the secret message. But in steganographic system the hacker has to extract the embedded from the host audio file. So it is a very difficult to break the system by hackers.

III. RELATED WORKS

A survey of steganographic techniques [15] reveals that there have been several techniques for hiding information or messages in host messages in such a manner that the embedded data should be imperceptible.

Substitution system [15] substitutes redundant parts of a hidden with a secret message. Spread spectrum techniques adopt ideas from spread spectrum communication [3]. The statistical method encodes information by changing several

statistical properties of a hidden and use hypothesis testing in the extraction process [3]. Distortion process stores information by signal distortion and measure the deviation from the original hidden in the decoding step [15]. The hidden generation method encodes information in the way a hidden for secret communication is created [3]. In case of hiding information in digital sound, phase Coding [16] embeds data by altering the phase in a predefined manner. To a certain extent, modifications of the phase of a signal cannot be perceived by the human auditory system (HAS) [6].

All these steganographic techniques deal with a few common types of steganography procedure depending on the variation of the host media. That means the hidden object [13] or the carrier object which will be used to hidden the secret data. Different media like image, text, video and audio has been used as a carrier or host media in different times [17]. Using audio file as a hidden object directs to Audio steganography. But in practical audio embedded systems face hard challenges in fulfilling all three requirements due to the large power and dynamic range of hearing, and the large range of audible frequency of the [1].

The human auditory system (HAS) perceives sounds over a range of power greater than 109:1 and a range of frequencies greater than 103:1. The sensitivity of the HAS to the Additive White Gaussian Noise (AWGN) is high as well; this noise in a sound file can be detected as low as 60 dB much below ambient level. On the other hand, opposite to its large dynamic range, HAS contains a fairly small differential range, i.e. loud sounds generally tend to mask out weaker sounds [18]. Additionally, HAS is insensitive to a constant relative phase shift in a stationary audio signal and some spectral distortions interprets as natural, perceptually non-annoying ones.

Two properties of the HAS dominantly used in steganographic techniques are frequency masking [18] and temporal masking [7]. The concept using the perceptual holes of the HAS is taken from wideband audio coding (e.g. MPEG compression 1, layer 3, usually called mp3) [19]. In the compression algorithms [7], the holes are used in order to decrease the amount of the bits needed to encode audio signal, without causing a perceptual distortion to the coded audio. On the other hand, in the information hiding scenarios, masking [18] properties are used to embed additional bits into an existing bit stream, again without generating audible noise in the audio sequence used for data hiding.

Some of the audio steganographic techniques are Lossless Adaptive Digital Audio Steganography [7], LSB based Audio Steganography [9], Audio Steganography using bit modification [8] etc.

IV. DESIGN METHODOLOGY

In the current endeavor, an audio file with ".wav" extension has been selected as host file. It is assumed that the least significant bits of that file should be modified without degrading the sound quality.

To do that, first one needs to know the file structure of the audio file. Like most files, WAV files have two basic parts, the header and the data. In normal wav files, the header is situated in the first 44 bytes of the file. Except the first 44 bytes, the rest of the bytes of the file are all about the data. The data is just one giant chunk of samples that represents the whole audio. While embedding data, one can't deal with the header section. That is because a minimal change in the header section leads to a corrupted audio file.

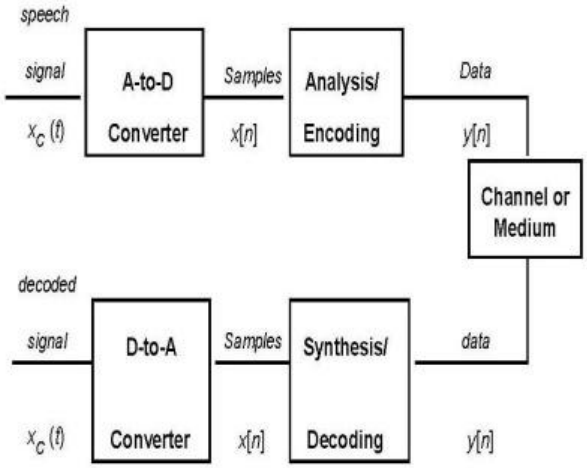

Figure 2. Audio Steganography

A program has been developed which can read the audio file bit by bit and stores them in a different file. The first 44 bytes should be left without any change in them because these are the data of the header section. Then start with the remaining data field to modify them to embed textual information. For example, if the word "Sonor" has to be embedded into an audio file one has to embed the binary values of the word "Sonor" into the audio data field. Consider the following table:

TABLE I. LETERS WITH ASCII VALUES AND CORRESPONDING BINARY VALUES

Letter	ASCII Value	Corresponding Binary Value
S	083	01010011
o	111	01101111
n	110	01101110
o	111	01101111
r	114	01110010

From the table, one can come to a point that to embed the word "Sonor" into the host audio file actually the corresponding eight bit binary values have to be embedded into the data field of that audio file.

The audio file with ".wav" will be selected as host file. The header and data be separated from the host file. Embedding of secret text be done to the data as header easily gets corrupted. The secret text be embedded at the LSB (Least Significant Bit) in alternative samples. By this the quality of audio file will not degrades and gets a stego audio. The above process is observed in fig. 3.

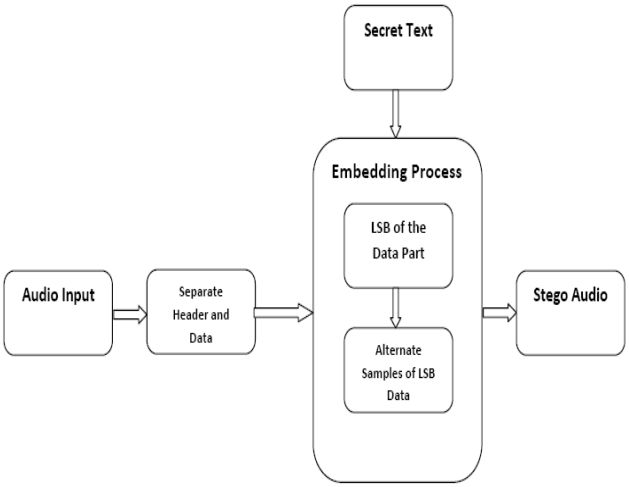

Figure 3. Stego Audio Process 1

The secret text be extracted by separating the data and header from the stored stego audio. The alternative samples of LSB of stego data file be left shifted to previous bits. The obtained binary be converted to ASCII and observed in text.

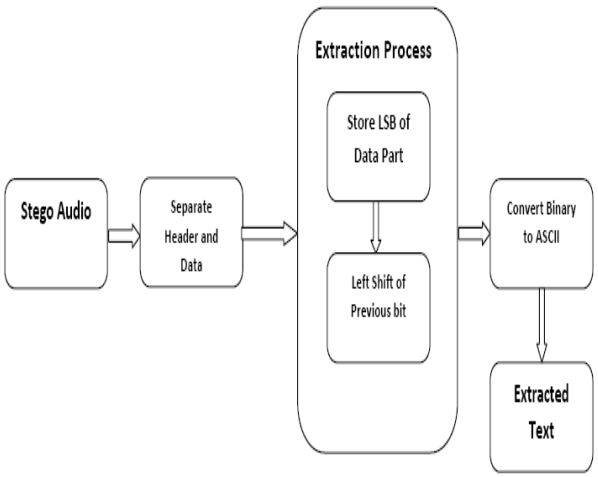

Figure 4. Stego Audio Process 2

V. ALGORITHM

The audio has been checked perfectly because a minimal change in To develop this algorithm multiple bits of each sample of the file have been changed or modified to insert text data in it. It has also been observed the degradation of the host audio file after modification of the bits. The bit modification was done by various ways, like 1, 2, 3, 4 bits were changed in turn. But after going through all the modification it has been observed that 1 bit change in LSB gave the best result. Thus, data can be embedded according to the following algorithm.

A. Algorithm (For Embedding of Data):

• Leave the header section of the audio file untouched...

• Start from a suitable position of the data bytes. (For the experiment purpose the present start byte was the 71st byte). Edit the least significant bit with the data that have to be embedded.

• Take every alternate sample and change the least significant bit to embed the whole message.

The data retrieving algorithm at the receiver's end follows the same logic as the embedding algorithm.

B. Algorithm (For Extracting of Data):

• Leave first 70 bytes.

• Start from the 71st byte and store the least significant bit in a queue.

• Check every alternate sample and store the least significant bit in the previous queue with a left shift of the previous bit.

• Convert the binary values to decimal to get the ASCII values of the secret message.

• From the ASCII find the secret message.

VI. EXPERIMENTATION, RESULTS AND INTERPRETATION

An audio file named "audio.wav" has been selected for this experiment. After checking the binary values of each sample, first 54 samples were left without any changes. The data embedding with LSB modification has been started after the header section. If the data embedding process is started from 71st sample then the LSB value of the 71st sample should be modified. If the binary value of the corresponding sample is "01110100" then "1" should be modified. From Table I it can be observed that to embed the letter "S", the sender has to embed the binary value "01010011". That is why according to the embedding algorithm "S" should be embedded according to Table II.

TABLE II. SAMPLES OF AUDIO FILE WITH BINARY VALUES BEFORE & AFTER EMBEDDING

Sample No.	Binary values of corresponding sample	Binary value to be embedded	Binary values After modification
71	01000111	0	01000110
73	01001001	1	01001001
75	01001011	0	01001010
77	01001101	1	01001101
79	01001111	0	01001110
81	01010001	0	01010000
83	01010011	1	01010011
85	01010101	1	01010101

According to the same way the remaining consecutive letters of the word "Sonor"is embedded in the file "audio.wav." Editing of the existing binary values with the intended binary values causes a minimal change in the audio file "audio.wav" that remains almost imperceptible to anyone other than the sender.When it comes to the point of data retrieving at the receiver's end, the retrieving algorithm has to be followed. First, change the audio message into binary format that has come from the source as stego-object. Leave first 70 bytes with no change in them.

Start from 71st bit, check the least significant bit, and store it in a queue. Check every alternate sample to collect the whole messages. Like 73rd, 75th and 77th and so on. Store the least significant bits of the alternate samples in the queue with left

shift of previous bit. Convert the binary values to decimal to get back the ASCII from which the text can be retrieved. The whole retrieval process can be depicted with the following table more thoroughly:

TABLE III. EXRACTION OF DATA FROM AUDIO FILE

Sample No.	Binary values with embedded secret data	Bits that are stored in the queue
71	01000110	0
73	01001001	01
75	01001010	010
77	01001101	0101
79	01001110	01010
81	01010000	010100
83	01010011	0101001
85	01010101	01010011

As in Table II the embedding process of the letter "S" was stated that is why, in Table III, the retrieval process of "S" is depicted. Starting from the 71st sample, every alternate sample has been checked and the least significant bit has been stored into a queue with a left shift of previous bit. After getting all the bits in the queue, start from the left hand side, take 8 bits and convert them into equivalent decimal to get the ASCII, from the ASCII retrieve the embedded textual message.

From the table, it is clearly observed that after getting "01010011"in the queue it is converted into the equivalent decimal that is 85, the ASCII of "S". Thus "S" is retrieved. Like the same way, the next letters also have been retrieved and hence the complete word "Audio."

VII. CONCLUSION

A method of embedding text-based data into a host audio file using the method of bit modification has been presented in this paper. A procedure has been developed in which the data field is edited to embed intended data into the audio file. To proceed with this, the header section of the header section may leads to a corruption of whole audio file.

In this algorithm, as an experiment first 70 bytes have been left untouched and starting from the 71st bytes every alternate sample has been modified to embed textual information. How the performance is affected by changing different bit fields has not been reported in this work. A study was made to see how the changing of a specific bit field creates degradation in the host audio file and in which point it leads to perceptible change in the audible sound quality to any other third party other than the sender or receiver. It was noticed that changing the least significant bit of the bytes gave the best results.

An audio file with a minimum size 700 KB be used. As the size of the audio file increases more data can be embedded into a host audio file. But time taken for embedding the data into a host audio file is time consuming. The maximum text file size that can be embedded in this audio file without degrading the file structure can be traced through survey. The main goal of this project was embedding of text into audio as a case of steganography. The two primary criteria for successful steganography are that the stego signal resulting from embedding is perceptually indistinguishable from the host audio signal, and the embedded message is re-hiddened correctly at the receiver. In test cases the text-based data has been successfully embedded to the audio file to visualize in what extent the target has been achieved.

REFERENCES

[1] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for data hiding", IBM Systems Journal, vol. 35, Issues 3&4, 1996, pp. 313-336.

[2] Kharrazi, M., Sencar, Husrev T., and Memon, N., "Image Steganography: Concepts and Practice", WSPC, April 22, 2004.

[3] Stefan Katzenbeisser, Fabien A. P. Petitcolas, "Information Hiding Techniques for Steganography and Digital Watermarking". Boston, Artech House, pp. 43 – 82. 2000.

[4] K. Matsui and K. Tanaka. Video-steganography. In: IMA Intellectual Property Project Proceedings, volume 1, pp 187-206, 1994.

[5] N.F. Johnson and S. Jajodia, "Exploring Steganography: Seeing the Unseen," Computer, vol. 31, no. 2, pp. 26-34, IEEE, Feb. 1998.

[6] Matsuoka, H., "Spread Spectrum Audio Steganography using Sub – band Phase Shifting", Proceedings of the 2006 International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIHMSP' 06), IEEE, 2006.

[7] S.S. Agaian, D. Akopian, O. Caglayan, S. A. D'Souza, "Lossless Adaptive Digital Audio Steganography," In Proc. IEEE Int. Conf. Signals, Systems and Computers, pp. 903-906, November 2005.

[8] K. Gopalan, "Audio steganography using bit modification", Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing, Vol. 2, pp. 421-424, April 2003.

[9] Mohammad Pooyan, Ahmed Delforouzi, "LSB-based Audio Steganography Method Based on Lifting WaveletTransform", International Symposium on Signal Processing and Information Technology, IEEE, 2007.

[10] C. C. Chang, T. S. Chen and H. S. Hsia, "An Effective Image Steganographic Scheme Based on Wavelet Transform and Pattern-Based Modification", IEEE Proceedings of the 2003 International Conference on Computer Networks and Mobile Computing, 2003.

[11] Chen and G.W. Womell, "Quantization index modulation: a class of provably good methods for digital watermarking and information embedding", IEEE Transactions on Information Theory, Vol. 47, No. 4, pp. 1423-1443, May 2001.

[12] B. Chen, "Design and analysis of digital watermarking, information embedding, and data hiding systems," Ph.D. dissertation, MIT, Cambridge, MA, June 2000.

[13] M. M. Amin, M. Salleh, S. Ibrahim, M. R. Katmin and M. Z. I. Shamsuddin, "Information Hiding using Steganography", 4th National Conference on Telecommunication Technology Proceedings, Shah Alam, Malaysia, IEEE, 2003.

[14] J. Zollner, H. Federrath, H. Klimant, et al., "Modelling the Security of Steganographic Systems", in 2nd Workshop on Information Hiding, Portland, April 1998, pp. 345-355.

[15] Johnson, Neil F. and Stefan Katzenbeisser. "A Survey of Steganographic Techniques", In Information Hiding: Techniques for Steganography and Digital Watermarking. Boston, Artech House. 43-78. 2000.

[16] Y.Yardimci, A.E.Cetin and R.Ansari, "Data hiding in speech using phase coding", ESCA. Eurospeech97, pp. 1679-1682, Greece, Sept. 1997.

[17] K. Bennett, "Linguistic Steganography: Survey, Analysis, and Robustness Concerns for Hiding Information in Text", Purdue University, CERIAS Tech. Report 2004-13, 2004.

[18] Matsuoka, H., "Spread Spectrum Audio Steganography using Sub – band Phase Shifting", Proceedings of the 2006 International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIHMSP'06), IEEE, 2006.

[19] Noll P., "Wideband speech and audio coding". IEEE Communications Magazine 31(11): 1993, pp 34–4